Non-linearities in Gaussian processes with integral observations

Tanskanen, Ville Elmeri

IEEE Computer Society
2020

Tanskanen , V E , Longi , K E & Klami , A 2020 , Non-linearities in Gaussian processes with integral observations . in 2020 IEEE 30th International Workshop on Machine Learning for Signal Processing (MLSP) . IEEE Computer Society , International Workshop on Machine Learning for Signal Processing , Espoo , Finland , 21/09/2020 . https://doi.org/10.1109/MLSP49062.2020.9231553

http://hdl.handle.net/10138/326061
https://doi.org/10.1109/MLSP49062.2020.9231553

acceptedVersion

NON-LINEARITIES IN GAUSSIAN PROCESSES WITH INTEGRAL OBSERVATIONS

Ville Tanskanen, Krista Longi, Arto Klami

Department of Computer Science, University of Helsinki

ABSTRACT

Gaussian processes (GP) can be used for inferring latent continuous functions also based on aggregate observations corresponding to integrals of the function, for example to learn daily rate of new infections in a population based on cumulative observations collected only weekly. We extend these approaches to cases where the observations correspond to aggregates of arbitrary non-linear transformations of a GP. Such models are needed, for example, when the latent function of interest is known to be non-negative or bounded. We present a solution based on Markov chain Monte Carlo with numerical integration for aggregation, and demonstrate it in binned Poisson regression and in non-invasive detection of fouling using ultrasound waves.

***Index Terms*—** Gaussian process, integral observation, aggregated data, non-negativity

1. INTRODUCTION

Gaussian processes (GP) provide a flexible basis for learning e.g. latent spatiotemporal functions based on noisy observations. In this work we concentrate in use of GPs to infer latent functions based on *aggregated observations* that relate to integrals of the function [1, 2, 3], also sometimes referred to as *binned data* [4]. Instead of directly observing a noisy realisation of the function itself, we observe a noisy average or sum of it over some, typically temporal or spatial, region. A prototypical application would be modeling a daily rate of incidences of a disease based on records of total number of new cases per week or month. In this context, GPs have been used for example to model malaria incidences and poverty rates on a finer scale based on data aggregated by administrative districts [1, 5], and computed tomography for reconstructing a 3D object based on signal attenuation along linear paths through the object [2, 6].

GPs can be used for modeling aggregated or binned data because they are closed under linear operators, including integration or finite summation. In other words, we retain analytic posterior distribution for the latent function even if conditioning on observations that are noisy integrals of the function.

This work was supported by the Academy of Finland Flagship programme: Finnish Center for Artificial Intelligence, FCAI, and grant 324852

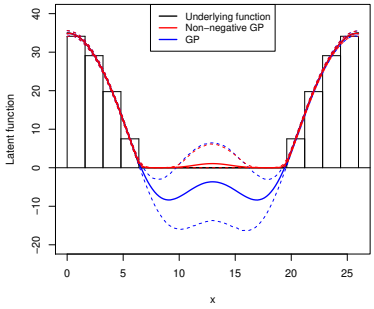

Fig. 1. Illustration of Gaussian process modeling of aggregated data when the latent-function is known to be non-negative. Each bar represents an observation of one aggregated value for the spanned horizontal region, and the red lines (dotted lines are 5% and 95% quantiles) represent the posterior of the latent function modeled as $\log(1 + e^{f(x)})$ for $f(x)$ with GP prior, correctly recovering the true latent function. The blue lines show how naively assuming the latent function to directly follow standard Gaussian process results in majority of the posterior mass to be on negative values in the middle region with no observations.

The posterior depends on similarity between each pair of aggregation supports and evaluating this requires numerical integration [1, 7, 8] or spectral approximation [6, 9] for general kernels and supports, and can only be computed analytically for restricted special cases [4, 10, 11]. For many cases (univariate intervals, line integrals for CT scans, spatial aggregation over convex areas) this is still computationally efficient.

As with GPs in general, we get analytic posterior expression only for observations corrupted by Gaussian noise, but approximate techniques such as variational approximation have been derived also for aggregate observation GPs to support other noise distributions [3, 5]. However, most of the existing works still make a very strong assumption that each observation corresponds to directly integrating a GP. That is, each noise-free observation must be of form $\int f(x)dx$ for some $f \sim GP$. For many – if not most – applications this simplification limits the accuracy of the models. For example, disease incidences are by definition non-negative, and the signal attenuation in computed tomography may

978-1-7281-6662-9/20/$31.00

depend on material properties in a non-linear manner. To model such phenomena accurately we would rather want to condition the GP on observations of the form $\int h(x)dx$ for some function $h(x)$ that satisfies the constraints imposed by the application, for example that $h(x) \geq 0 \; \forall \, x$. Smith et al. [4] attempted imposing such constraints by introducing imaginary pseudo-observations designed to enforce positivity constraint, and Law et al. [5] proposed a variational approximation that supports some particular constraints for specific noise assumptions, but no general solutions are available. Figure 1 illustrates the concept by demonstrating how non-negativity constraint influences the latent function when modeling binned data, similar to the example of [4].

In this work we present a solution for learning the GP posterior when conditioned on integral observations of the form $\int g(f(x))dx$, where $g(\cdot)$ is an arbitrary non-linear function selected so that $g(f(x))$ matches the application needs. For example, for non-negative rates we can choose $g(x) = e^x$ or $g(x) = \log(1 + e^{f(x)})$ (softplus), and known non-linearities in signal attenuation can be incorporated by selecting $g(x)$ based on physical prior knowledge. This formulation is naturally not amenable to analytic calculation, but we present a general solution building on Hamiltonian Monte Carlo (HMC) sampling [12], implemented using the `Stan` probabilistic programming language [13] and using numerical quadratures for evaluating the integrals. The solution supports arbitrary functions $g(x)$, allows for full Bayesian inference for hyper-parameters (e.g. kernel length-scales), and works with all kernels.

We demonstrate the method in modeling count data with different aggregation areas, and use it to improve the accuracy in recent application of non-invasive fouling detection based on ultrasound wave propagation [11, 14] by accounting for the obvious physical constraint of non-negative fouling thickness. This reduces the estimation error by more than 40%.

2. GPS WITH INTEGRAL OBSERVATIONS

We consider problems where the goal is to estimate a latent function based on a collection of observations that are integrals of the function, also sometimes referred to as aggregated data or binned data. Formally, we assume that there is an underlying function $f : \mathcal{X} \to \mathbb{R}$ which generates samples for a region $v \subset \mathcal{X}$ through a probability distribution $y|f, v \sim p\left(y|\int_v g(f(x))dx\right)$, where $g(\cdot)$ is typically non-linear function that is assumed to be known based on domain knowledge. A single observation is a pair (v_i, y_i), where the v_i is a subset of the input space $\mathcal{X}$, referred as *region* in this work, and y_i is the value of the response variable for that region. When a collection of the region-response pairs $(v_i, y_i)_{i=1}^N$ is observed, our goal is to recover the function $f(\cdot)$ which generated these samples. A collection of N observed regions and response variable values are denoted by $\boldsymbol{v} = [v_1, \ldots, v_N]$ and $\boldsymbol{y} = [y_1, \ldots, y_N]$ respectively. In this work we focus on regions of regular shapes, intervals in 1D feature space and lines in 2D feature space, but there are no limitations that prevent the use of arbitrary shapes.

Gaussian processes provide flexible priors for unknown functions [15]. We use a GP prior to describe our beliefs about the smoothness, magnitude and rate of change of the unknown function. Formally, GP is a (possibly infinite) set of random variables for which any finite linear combination follows a Gaussian distribution, and the smoothness is determined by a positive definite covariance function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$. The covariance matrix between random vectors $\boldsymbol{a} \in \mathbb{R}^M$ and $\boldsymbol{b} \in \mathbb{R}^N$, at collection of points $\boldsymbol{X_a}$ and $\boldsymbol{X_b}$ respectively, is denoted by $K_{\boldsymbol{ab}} \in \mathbb{R}^{M \times N}$, which is computed between each element in $\boldsymbol{X_a}$ and $\boldsymbol{X_b}$.

The goal is to recover the posterior distribution

$$p(f|\boldsymbol{v}, \boldsymbol{y}) \propto p(\boldsymbol{y}|f, \boldsymbol{v})p(f|\boldsymbol{v}), \tag{1}$$

where $p(f|\boldsymbol{v})$ is given a GP prior and $p(\boldsymbol{y}|f, \boldsymbol{v})$ is the likelihood of the observed data given f, and whose functional form is dependent on the type of the response variable y (see Section 3 for concrete example for Poisson likelihood). We use f to denote the unknown function as a mathematical object and $\boldsymbol{f_a}$ to denote the vector of function values evaluated at vector $\boldsymbol{a}$. In the likelihood we assume that the response variables are conditionally independent given the latent function values at the given region. This allows us to factorize the likelihood as

$$p(\boldsymbol{y}|f, \boldsymbol{v}) = \prod_{i=1}^{N} p\left(y_i|\int_{v_i} g(f(\boldsymbol{x}))d\boldsymbol{x}\right).$$

3. METHODS

In the following, we present the technical details for learning the posterior (1) for cases with both arbitrary likelihood function $p(y|\cdot)$ and arbitrary non-linear transformation $g(\cdot)$ applied before integration. Note that for linear $g(\cdot)$ there are also other inference techniques; see Section 4.

3.1. Inference

The inference is done using a variant of HMC called the No-U-Turn Sampler (NUTS) [12] with `Stan` probabilistic programming language [13]. The integrals, appearing in the likelihood term, are computed using numeric quadrature by evaluating the value of $g(f(x))$ for some collection of M points x in v_i. In this work we use Riemann sums

$$\int_{v_i} g(f(x))dx \approx \sum_{j=1}^{M} \Delta_{v_i} g(f(x_{ij}))$$

for evenly-spaced grid $x_{ij} \in \mathbf{v}_i := [x_{i1}, \ldots, x_{iM}]$ discretizing the observed region v_i, but note that the strategy can also be used with more advanced quadratures or other discretization schemes.

To compute all of the required likelihoods we represent the posterior for the NM -dimensional set of points

$$\mathbf{v} := [\mathbf{v}_1, \ldots, \mathbf{v}_N] = [x_{11}, \ldots, x_{1M}, \ldots, x_{N1}, \ldots, x_{NM}]$$

collecting all of the discretization locations into one vector, and we denote the sth sample of this posterior representation by $\boldsymbol{f}_{\mathbf{v}}^{(s)}$. When we wish to acquire the distribution of $\boldsymbol{f}_{\boldsymbol{x}}$ at some locations $\boldsymbol{x} \subset \mathcal{X}$, not included in $\mathbf{v}$, we employ the samples $\{\boldsymbol{f}_{\mathbf{v}}^{(s)}\}_{s=1}^{S}$ of the posterior. For each sample $\boldsymbol{f}_{\mathbf{v}}^{(s)}$ the distribution of $\boldsymbol{f}_{\boldsymbol{x}}$ is Gaussian

$$\boldsymbol{f}_{\boldsymbol{x}}|\boldsymbol{f}_{\mathbf{v}}^{(s)} \sim N(K_{\boldsymbol{x}\mathbf{v}}K_{\mathbf{v}\mathbf{v}}^{-1}\boldsymbol{f}_{\mathbf{v}}^{(s)}, K_{\boldsymbol{x}\boldsymbol{x}} - K_{\boldsymbol{x}\mathbf{v}}K_{\mathbf{v}\mathbf{v}}^{-1}K_{\boldsymbol{x}\mathbf{v}}^{T}), \quad (2)$$

and the collection of S samples determines the full posterior representation.

Typical covariance functions contain parameters, often called hyperparameters, controlling for example the rate of change and the magnitude of $f(\cdot)$. Since we anyway sample directly the latent function values $\boldsymbol{f}_{\mathbf{v}}$, we can easily carry out posterior inference over the hyperparameters as well by including them as part of the representation. As a practical note, however, this requires computing the covariance matrix within the `Stan` model, based on precomputed distances between the elements of $\mathbf{v}$, whereas for fixed hyperparameters the whole kernel can be precomputed. Hence, for computational efficiency one might also want to optimize the hyperparameters by maximizing the marginal likelihood [15].

3.2. Example: Poisson regression

To better illustrate the technical definition above, we next demonstrate how the general approach can be used for modeling binned count-valued observations with non-negative underlying rate function [4, 16]. More precisely, we consider a case where $\mathcal{X}$ is discrete (each x corresponding e.g. to a day) and the goal is to learn the rate for each x based on aggregates over finite set of elements (e.g. week).

We begin by first assuming that a single observation at location x follows a Poisson distribution $y|f, x \sim Poi(g(f(x)))$ conditionally independent of other observations. Given the conditional independence it directly follows that the aggregated count for a region v_i follows

$$y_i|f, v_i \sim Poi\left(\sum_{x \in v_i} g(f(x))\right), \quad (3)$$

which is now a sum instead of integral due to the discrete $\mathcal{X}$.

To ensure positive rate for the Poisson distribution we need $g : \mathbb{R} \to \mathbb{R}^+$, and in this work we use both $g(x) = \exp(x)$ and softplus $g(x) = \log(1 + \exp(x))$ as examples. See e.g. [16] for discussion on the implications of the choice. Note that with identity link $g(x) = x$ the only way to achieve positive rate would be to use high prior mean, which would severely bias results especially for small observed counts.

3.3. Inducing points

Expressing the posterior requires inference over the NM-dimensional set of points $\mathbf{v}$, where N is the number of observed regions and M is the number of discretization points, and hence the kernel matrix is of size $\mathbb{R}^{NM \times NM}$. For many applications, such as most physical sensor settings where data collection is laborious [14], the amount of observations is small and the computation remains efficient. With larger data sets the problem can be alleviated by using inducing points at the level of the underlying latent process [17], as has previously been demonstrated for integral observations in the context of variational inference [3, 5]. Here we describe how inducing points can be used in our context to speed up computation for problems with large N.

Instead of directly performing inference over $\boldsymbol{f}_{\mathbf{v}} \in \mathbb{R}^{NM}$, we perform inference for the function values $\boldsymbol{f}_{\boldsymbol{u}}$ for some N_{ind} inducing points $\boldsymbol{u} \subset \mathcal{X}$ that are fixed locations of the input space. To compute the posterior density (1) we now need to infer $\boldsymbol{f}_{\mathbf{v}}$ using $\boldsymbol{f}_{\boldsymbol{u}}$ using

$$\log p(\boldsymbol{f}_{\boldsymbol{u}}|\mathbf{v}, \boldsymbol{u}, \boldsymbol{y}) = \log p(\boldsymbol{y}|\boldsymbol{f}_{\boldsymbol{u}}, \mathbf{v}, \boldsymbol{u}) + \log p(\boldsymbol{f}_{\boldsymbol{u}}|\boldsymbol{u}) + C, \quad (4)$$

where the Gaussian process prior is given to $p(\boldsymbol{f}_{\boldsymbol{u}}|\boldsymbol{u})$, C is a constant, and the log-likelihood becomes

$$\begin{aligned} \log p(\boldsymbol{y}|\boldsymbol{f}_u, \mathbf{v}, \boldsymbol{u}) &= \sum_{i=1}^{N} \log \int p(y_i, \boldsymbol{f}_{\mathbf{v}_i}|\boldsymbol{f}_u, \mathbf{v}, \boldsymbol{u}) d\boldsymbol{f}_{\mathbf{v}_i} \\ &= \sum_{i=1}^{N} \log \mathbb{E}_{\boldsymbol{f}_{\mathbf{v}_i}|\boldsymbol{f}_{\boldsymbol{u}}, \mathbf{v}_i, \boldsymbol{u}} \left[p(y_i|\boldsymbol{f}_{\mathbf{v}_i}, \mathbf{v}_i)\right]. \end{aligned}$$

Note that the random variable $\boldsymbol{f}_{\mathbf{v}_i}|\boldsymbol{f}_{\boldsymbol{u}}, \mathbf{v}_i, \boldsymbol{u}$ follows Gaussian distribution [15]

$$\boldsymbol{f}_{\mathbf{v}_i}|\boldsymbol{f}_{\boldsymbol{u}}, \mathbf{v}_i, \boldsymbol{u} \sim N(\underbrace{K_{\mathbf{v}_i u}K_{uu}^{-1}\boldsymbol{f}_u}_{:=\mu_{\boldsymbol{f}_{\mathbf{v}_i}}}, \Sigma_{\boldsymbol{f}_{\mathbf{v}_i}}).$$

Finally, we approximate the joint likelihood using

$$\log p(\boldsymbol{y}|\boldsymbol{f}_{\boldsymbol{u}}, \mathbf{v}, \boldsymbol{u}) \approx \sum_{i=1}^{N} \log p(y_i|\mu_{\boldsymbol{f}_{\mathbf{v}_i}}, \mathbf{v}, \boldsymbol{u}),$$

where first order Taylor approximation at $\mu_{\boldsymbol{f}_{\mathbf{v}_i}}$ was used for the density $p(y_i|\boldsymbol{f}_{\mathbf{v}_i}, \mathbf{v}_i)$. In our experiments the first order approximation worked well, but we note that also higher order series or Monte Carlo approximation could be used.

4. RELATED WORK

Aggregated data are being observed in many applications such as in modeling air pollution [3] and infectious diseases [5]. Methods for modeling aggregated data have been extensively studied in e.g. geostatistics, where learning more fine-scaled estimates from aggregated spatial data is

known as *downscaling*, *disaggregation* or *change of support* [18, 19, 20]. In machine learning, models for different forms of aggregated data have been studied under *multiple instance learning*, *learning from label proportions* and *learning on aggregate outputs* [21, 22], and recent works have also presented GP-based approaches to these problems [5, 23, 24].

Our work is on specific type of aggregated data, where the input corresponds to a region and the output corresponds to an integral over this, like defined by Kyriakidis [7], and next we discuss the most closely related GP methods for such setups. Law et al. [5] proposed the bag observation model for aggregation over continuous spatial regions, using variational approximation for inference for exponential family models, matching our general formulation but having more limited scope. Binned data can be seen as integral observations [4], and recently continuous supports have also been considered in multi-task learning, where the goal is to learn the latent function based on data sets that have been aggregated at different input scales [1, 3, 25]. Integral observations naturally occur also in many physical sensing applications, like laser scanners [8], tomographic reconstruction [2, 6] and ultrasonic sensing [11, 14], where the work based on GPs has been done mostly independently of the work on aggregated data.

Majority of the earlier work is limited to conjugate cases, where inference can be carried out directly on the level of the integral observations, as opposed to point-wise observations as in our work. While this results in smaller kernel matrix in $\mathbb{R}^{N\times N}$ that collects double integrals of the kernel function, many of the earlier works compute these integrals numerically [1, 3, 7, 8] and require discretization similar to ours. For specific kernels and region shapes faster algorithms have been proposed based on spectral approximations [2, 6, 9] and fully or partially analytic integration [4, 10, 11].

Some works have also considered non-conjugate scenarios, by supporting more general likelihoods or by supporting non-linear transformations. Smith et al. [4] enforced non-negativity for modeling binned data by introducing virtual observations solely for constraining the latent function, whereas Yousefi et al. [3] and Law et al. [5] proposed variational approximations for exponential family likelihoods that were demonstrated also in applications with additional constraints on parameters. However, [3] only considers non-linearities after the aggregation, and [5] only supports specific computationally tractable transformations for specific likelihoods.

We provide the fist general solution for accounting arbitrary non-linear transformations before aggregation. The proposed method is less efficient than the variational approximations [3, 5], but for many applications this is not a practical bottleneck. For example, in physical sensing problems the number of observations is kept small due to the cost of installing sensors. In this work we only considered integral observations, but it is likely that the proposed approach can be extended also for more general cases of aggregated data, such as multiple instance learning setups.

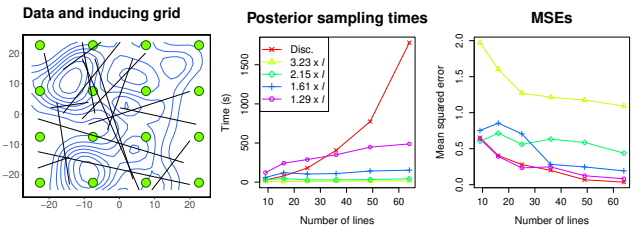

Fig. 2. HMC sampling times (middle) and mean-squared errors (MSE) (right) for problems of different number of observations (x-axis) and varying number of inducing points (colors). For sufficiently dense grid of inducing points (purple and blue) we get almost exact estimates with lower computational cost, but with too few inducing points relative to the smoothness of the function the results can be bad. These results are provided for the function illustrated in the left sub-plot, which also shows an example inducing point grid for spacing 2.15 times the length-scale and 16 random integral observation regions (black lines), but would be similar for other functions.

5. EXPERIMENTS

We start by demonstrating the use of inducing points to speed up computation, and then apply the proposed methodology for two applications that require use of non-linear transformation $g(\cdot)$. The code for replicating the experiments is available at `https://version.helsinki.fi/MUPI/mlsp2020`.

5.1. Inducing points

Figure 2 illustrates the effect of using inducing points, by plotting the computational time and accuracy for different number of observations (N) and inducing points (N_{ind}) for an example function sampled from a GP prior and transformed with softplus before computing the integrals along randomly sampled line segments $\boldsymbol{v}$. The inducing points were placed at regular grids, and for inference we used the same kernel (squared exponential) with correct hyperparameters that were used for the true function, to focus on illustrating the effect of the inducing points.

For small N, direct sampling of all NM discretization points is to be preferred, but for larger N using inducing points clearly reduces the computational cost. For sufficiently fine grid the result is still essentially as accurate (measured here using mean-square error between the true function and the mean estimate), but with too coarse grid, where the gap between the neighboring inducing points is considerably longer than the kernel length-scale, the accuracy naturally drops. The grid spacing is expressed in terms of the length-scale, so that the results can be translated also for other kernels and input domains.

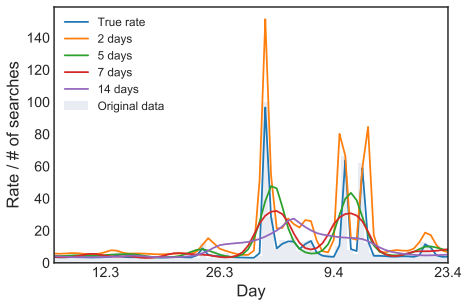

Fig. 3. Estimating the rate of daily Google search counts based on count data aggregated with different granularities (2-14 days). The true rate was estimated using the original non-binned data.

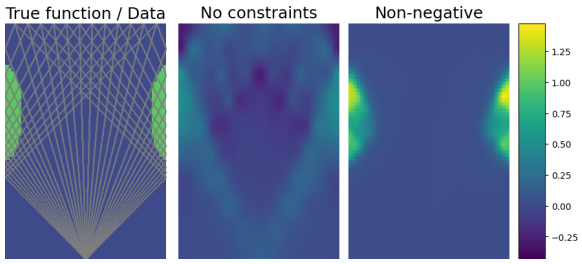

Fig. 4. **Left:** The measurement setup provides one observation integrating the underlying fouling thickness along each of the lines depicted here, overlaid on top of the ground truth function. **Middle:** Without non-negativity constraint the mean estimate is non-zero for large areas. **Right:** Non-negativity constraint helps localizing the true fouling.

5.2. Poisson regression

We demonstrate Poisson regression presented in section 3.2 on time series count data (Figure 3). As an example, we use the daily Google search counts of term "earthquake" in the U.S for the last 70 days. The daily data is aggregated using different granularities (2-14 days), and the goal is to learn the true daily rate (estimated from raw data without aggregation) based on these aggregate observations alone. We use the squared exponential kernel, performing posterior inference over its hyperparameters, and the exponential transformation $g(f(x)) = e^{f(x)}$. The general trend is correctly modeled with all granularities of aggregation, but sudden changes in rate are naturally smoothed when only observing heavily aggregated data.

5.3. Ultrasonic fouling detection

We demonstrate the usefulness of a non-negativity constraint in ultrasonic localization of fouling in closed metal pipe, following the simulated experiment of [14]. They interpret time-of-flight differences between clean and fouled pipe as integral observation along a flight path between transmitter and receiver, and for cylinder structures can record multiple helical paths between sensors. Figure 4 (left) shows flattened pipe surface that wraps around the edges, and lines are the paths from the transmitter at the bottom to receivers at the top.

The amount of fouling on a surface is naturally non-negative, which was ignored in [14] to retain conjugacy. We replicate their simulation experiment using the same kernel (Matern $3/2$) and hyperparameters ($l = 5$, $\sigma_f = 1$, $\sigma_\epsilon = 1$) they reported, but add the non-negativity constraint by using softplus-transformation for $f(\cdot)$ before integration. Figure 4 shows the mean of the GP fit with identity link corresponding to no constraint (middle) and with the non-negativity constraint (right). This simple addition reduces the root mean square error between the mean estimate and the true fouling function from 0.19 to 0.11.

6. CONCLUSION

Learning latent functions based on data that is only available at coarse rate or can otherwise be interpreted as arising from integration of the function is prevalent problem in spatial statistics and signal processing, and GPs provide theoretically strong and computationally convenient basis for this. Despite the flexibility of accounting for various types of aggregation areas and support for arbitrary likelihoods via approximations, these approaches are not applicable to scenarios where the latent function being integrated is not Gaussian but need to satisfy additional constraints, such as non-negativity. We presented the first general approach for addressing this, building on easy-to-use probabilistic programming formulation that allows for arbitrary non-linear transformations of the latent function, extending the preliminary works only applicable for limited transformations [3, 5] or using heuristic additional constraints [4]. While our approach is computationally less efficient and not directly applicable for large geospatial applications like in [1, 5], it is perfectly adequate for wide range of applications.

The flexible formulation is particularly beneficial in applications with rich prior information on properties of the system being modeled. In this work we demonstrated how incorporating already a very simple physical constraint of non-negativity in ultrasonic detection of fouling [14] reduced localization error by more than 40%, but the accuracy could be further improved by incorporating additional existing physical knowledge of the non-linearities involved in ultrasound propagation. The proposed approach allows plugging in arbitrary functions that need not even have closed-form expression but could be estimated from e.g. finite element method simulations of fouled structures. Extensions and generalizations like this would be extremely tedious – if not impossible – to derive for the competing approximate solutions.

References

[1] Yusuke Tanaka, Toshiyuki Tanaka, Tomoharu Iwata, Takeshi Kurashima, Maya Okawa, Yasunori Akagi, and Hiroyuki Toda, "Spatially aggregated Gaussian processes with multivariate areal outputs," in *NeurIPS*, 2019.

[2] Zenith Purisha, Carl Jidling, Niklas Wahlström, Thomas B Schön, and Simo Särkkä, "Probabilistic approach to limited-data computed tomography reconstruction," *Inverse Problems*, vol. 35, no. 10, 2019.

[3] Fariba Yousefi, Michael T Smith, and Mauricio Álvarez, "Multi-task learning for aggregated data using Gaussian processes," in *NeurIPS*, 2019.

[4] Michael Th Smith, Mauricio A Alvarez, and Neil D Lawrence, "Gaussian process regression for binned data," *arXiv:1809.02010*, 2018.

[5] Ho Chung Law, Dino Sejdinovic, Ewan Cameron, Tim Lucas, Seth Flaxman, Katherine Battle, and Kenji Fukumizu, "Variational learning on aggregate outputs with Gaussian processes," in *NeurIPS*, 2018.

[6] Carl Jidling, Johannes Hendriks, Niklas Wahlström, Alexander Gregg, Thomas B Schön, Christopher Wensrich, and Adrian Wills, "Probabilistic modelling and reconstruction of strain," *Nuclear Instruments and Methods in Physics Research Section B*, 2018.

[7] Phaedon C Kyriakidis, "A geostatistical framework for area-to-point spatial interpolation," *Geographical Analysis*, vol. 36, no. 3, pp. 259–289, 2004.

[8] Simon Timothy O'Callaghan and Fabio T Ramos, "Continuous occupancy mapping with integral kernels," in *AAAI*, 2011.

[9] Matthew Adelsberg and Christian Schwantes, "Binned kernels for anomaly detection in multi-timescale data using Gaussian processes," in *KDD 2017 Workshop on Anomaly Detection in Finance*, 2018, pp. 102–113.

[10] Johannes N Hendriks, Carl Jidling, Adrian Wills, and Thomas B Schön, "Evaluating the squared-exponential covariance function in Gaussian processes with integral observations," *arXiv 1812.07319*, 2018.

[11] Krista Longi, Chang Rajani, Tom Sillanpää, Joni Mäkinen, Timo Rauhala, Ari Salmi, Edward Hæggström, and Arto Klami, "Sensor placement for spatial Gaussian processes with integral observations," in *UAI*, 2020.

[12] Matthew D Hoffman and Andrew Gelman, "The no-u-turn sampler: adaptively setting path lengths in Hamiltonian Monte Carlo," *JMLR*, vol. 15, no. 1, 2014.

[13] Bob Carpenter, Andrew Gelman, Matthew Hoffman, Daniel Lee, Ben Goodrich, Michael Betancourt, Marcus Brubaker, Jiqiang Guo, Peter Li, and Allen Riddell, "Stan: A probabilistic programming language," *Journal of Statistical Software*, 2017.

[14] Tom Sillanpää, Timo Rauhala, Joni Mäkinen, Chang Rajani, Krista Longi, Arto Klami, Ari Salmi, and Edward Hæggström, "Ultrasonic fouling detector powered by machine learning," in *IUS*. IEEE, 2019.

[15] Carl Edward Rasmussen and Christopher KI Williams, *Gaussian processes for machine learning*, MIT press Cambridge, MA, 2006.

[16] Chris Lloyd, Tom Gunter, Michael Osborne, and Stephen Roberts, "Variational inference for Gaussian process modulated Poisson processes," in *ICML*, 2015.

[17] Mauricio Alvarez and Neil D Lawrence, "Sparse convolved Gaussian processes for multi-output regression," in *NeurIPS*, 2009.

[18] Jingxiong Zhang, Peter Atkinson, and Michael F Goodchild, *Scale in spatial information and analysis*, CRC Press, 2014.

[19] Carol A Gotway and Linda J Young, "Combining incompatible spatial data," *Journal of the American Statistical Association*, vol. 97, no. 458, pp. 632–648, 2002.

[20] António Xavier, Maria de Belém Costa Freitas, Maria do Socorro Rosário, and Rui Fragoso, "Disaggregating statistical data at the field level: An entropy approach," *Spatial Statistics*, vol. 23, pp. 91–108, 2018.

[21] David R Musicant, Janara M Christensen, and Jamie F Olson, "Supervised learning by training on aggregate outputs," in *ICDM*. IEEE, 2007, pp. 252–261.

[22] Hendrik Kück and Nando de Freitas, "Learning about individuals from group statistics," in *UAI*, 2005.

[23] Minyoung Kim and Fernando De la Torre, "Gaussian processes multiple instance learning," in *ICML*, 2010.

[24] Manuel Haußmann, Fred A Hamprecht, and Melih Kandemir, "Variational Bayesian multiple instance learning with Gaussian processes," in *CVPR*, 2017.

[25] Oliver Hamelijnck, Theodoros Damoulas, Kangrui Wang, and Mark Girolami, "Multi-resolution multi-task Gaussian processes," in *NeurIPS*, 2019.